NPS-AM-12-C9P06R01-053

EXCERPT FROM THE

PROCEEDINGS

OF THE

NINTH ANNUAL ACQUISITION RESEARCH SYMPOSIUM WEDNESDAY SESSIONS VOLUME I

A Framework for Reuse in the DoN

Randy Mactal
Space and Naval Warfare Systems Center Pacific
Lynne Spruill
APEO Engineering Support

Published April 30, 2012

Approved for public release; distribution is unlimited.
Prepared for the Naval Postgraduate School, Monterey, CA 93943.

Report Documentation Page	*Form Approved* *OMB No. 0704-0188*

Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
30 APR 2012		**00-00-2012 to 00-00-2012**

4. TITLE AND SUBTITLE **A Framework for Reuse in the DoN**	5a. CONTRACT NUMBER 5b. GRANT NUMBER 5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER 5e. TASK NUMBER 5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Space and Naval Warfare Systems Center Pacific,53560 Hull Street,San Diego,CA,92152-5001**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S) 11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT

Reuse offers the possibility of increasing engineering productivity, efficiency, and software quality while simultaneously reducing the cost of building software-intensive systems. The application of reuse has been around for many years and the DoD has made concerted efforts to implement reuse strategies since the early 90s. Although there are many excellent examples of its implementation throughout the Navy, efforts to implement software reuse strategies at an enterprise level have not matured enough to reap large-scale benefits. In the current fiscal climate of budget reductions and mandates for efficiencies, changes in acquisition, engineering, and business processes will require an enterprise reuse strategy that provides clear guidance, incentives, and compliance mechanisms. This paper discusses the current state of reuse within the DoN and proposes an implementation framework for a strategy-driven reuse approach.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			
unclassified	**unclassified**	**unclassified**	**Same as Report (SAR)**	**46**	

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

The research presented at the symposium was supported by the acquisition chair of the Graduate School of Business & Public Policy at the Naval Postgraduate School.

To request defense acquisition research or to become a research sponsor, please contact:

NPS Acquisition Research Program
Attn: James B. Greene, RADM, USN, (Ret.)
Acquisition Chair
Graduate School of Business and Public Policy
Naval Postgraduate School
Monterey, CA 93943-5103
Tel: (831) 656-2092
Fax: (831) 656-2253
E-mail: jbgreene@nps.edu

Copies of the Acquisition Research Program's sponsored research reports may be printed from our website (www.acquisitionresearch.net).

Preface & Acknowledgements

Welcome to our Ninth Annual Acquisition Research Symposium! This event is the highlight of the year for the Acquisition Research Program (ARP) here at the Naval Postgraduate School (NPS) because it showcases the findings of recently completed research projects—and that research activity has been prolific! Since the ARP's founding in 2003, over 800 original research reports have been added to the acquisition body of knowledge. We continue to add to that library, located online at www.acquisitionresearch.net, at a rate of roughly 140 reports per year. This activity has engaged researchers at over 60 universities and other institutions, greatly enhancing the diversity of thought brought to bear on the business activities of the DoD.

We generate this level of activity in three ways. First, we solicit research topics from academia and other institutions through an annual Broad Agency Announcement, sponsored by the USD(AT&L). Second, we issue an annual internal call for proposals to seek NPS faculty research supporting the interests of our program sponsors. Finally, we serve as a "broker" to market specific research topics identified by our sponsors to NPS graduate students. This three-pronged approach provides for a rich and broad diversity of scholarly rigor mixed with a good blend of practitioner experience in the field of acquisition. We are grateful to those of you who have contributed to our research program in the past and hope this symposium will spark even more participation.

We encourage you to be active participants at the symposium. Indeed, active participation has been the hallmark of previous symposia. We purposely limit attendance to 350 people to encourage just that. In addition, this forum is unique in its effort to bring scholars and practitioners together around acquisition research that is both relevant in application and rigorous in method. Seldom will you get the opportunity to interact with so many top DoD acquisition officials and acquisition researchers. We encourage dialogue both in the formal panel sessions and in the many opportunities we make available at meals, breaks, and the day-ending socials. Many of our researchers use these occasions to establish new teaming arrangements for future research work. In the words of one senior government official, "I would not miss this symposium for the world as it is the best forum I've found for catching up on acquisition issues and learning from the great presenters."

We expect affordability to be a major focus at this year's event. It is a central tenet of the DoD's Better Buying Power initiatives, and budget projections indicate it will continue to be important as the nation works its way out of the recession. This suggests that research with a focus on affordability will be of great interest to the DoD leadership in the year to come. Whether you're a practitioner or scholar, we invite you to participate in that research.

We gratefully acknowledge the ongoing support and leadership of our sponsors, whose foresight and vision have assured the continuing success of the ARP:

- Office of the Under Secretary of Defense (Acquisition, Technology, & Logistics)
- Director, Acquisition Career Management, ASN (RD&A)
- Program Executive Officer, SHIPS
- Commander, Naval Sea Systems Command
- Program Executive Officer, Integrated Warfare Systems
- Army Contracting Command, U.S. Army Materiel Command
- Office of the Assistant Secretary of the Air Force (Acquisition)

- Office of the Assistant Secretary of the Army (Acquisition, Logistics, & Technology)
- Deputy Director, Acquisition Career Management, U.S. Army
- Office of Procurement and Assistance Management Headquarters, Department of Energy
- Director, Defense Security Cooperation Agency
- Deputy Assistant Secretary of the Navy, Research, Development, Test & Evaluation
- Program Executive Officer, Tactical Aircraft
- Director, Office of Small Business Programs, Department of the Navy
- Director, Office of Acquisition Resources and Analysis (ARA)
- Deputy Assistant Secretary of the Navy, Acquisition & Procurement
- Director of Open Architecture, DASN (RDT&E)
- Program Executive Officer, Littoral Combat Ships

We also thank the Naval Postgraduate School Foundation and acknowledge its generous contributions in support of this symposium.

James B. Greene Jr. Rear Admiral, U.S. Navy (Ret.)	Keith F. Snider, PhD Associate Professor

Panel 6. Considerations in Acquiring Open Architecture Software Systems

Wednesday, May 16, 2012	
1:45 p.m. – 3:15 p.m.	**Chair: Captain Joseph J. Beel**, USN, Commanding Officer, Space and Naval Warfare Systems Center Pacific ***A Framework for Reuse in the DoN*** Randy Mactal, *Space and Naval Warfare Systems Center Pacific* Lynne Spruill, *APEO Engineering Support* ***Addressing Challenges in the Acquisition of Secure Software Systems With Open Architectures*** Walt Scacchi and Thomas Alspaugh *University California, Irvine* ***Certifying Tools for Test Reduction in Open Architecture*** Valdis Berzins, *Naval Postgraduate School*

Joseph J. Beel—Captain Joe Beel was commissioned from the U.S. Naval Academy in 1985, earning a Bachelor of Science degree in mechanical engineering. He was designated a Naval Aviator in September 1986. He completed Fleet Replacement Pilot training with HSL-31 in May 1987 and joined the Sea Snakes of HSL-33, flying the SH-2F Sea Sprite until December 1989. He deployed in the USS *Kirk* (FF1067), the USS *Knox* (FF 1052), the USS *Francis Hammond* (FF1067), and the USS *Sterrett* (CG 31), including service in Operation Earnest Will.

He attended the Naval Postgraduate School in Monterey, CA, from 1990 until 1992, earning a Master of Science (with distinction) in operations research. He taught in the U.S. Naval Academy Mathematics Department from May 1992 until May 1995 and served as the Fifth Company Officer from August 1993 until May 1995. He also served as an advanced seamanship and navigation instructor and was designated a craftmaster/yard patrol craft officer-in-charge afloat.

Captain Beel completed Fleet Replacement Pilot training with HSL-41 in February 1996 and joined the Battle Cats of HSL-43, flying the SH-60B Sea Hawk until 1998. He deployed in the USS *Princeton* (CG 59).

From June 1998 until August 1999, Captain Beel served as the training and education program analyst in the Assessment Division (N81), Office of the Chief of Naval Operations. He served in a Federal Executive Fellowship at the RAND Corporation in Santa Monica, CA, from August 1999 to August 2000. From August 2000 until September 2002, he served in the USS *John C. Stennis* (CVN 74), including service in Operations Noble Eagle and Enduring Freedom. He served as officer-in-charge of Navy Warfare Development Command, Detachment San Diego, from October 2002 until August 2003. He served as commanding officer and executive officer, Naval Air Technical Data and Engineering Service Command (NATEC), from September 2003 until September 2006.

Most recently, Captain Beel served four years in the Program Executive Office (PEO), Command, Control, Communication, Computers, and Intelligence (C4I); as PEO chief of staff and deputy for Operations from October 2006 to June 2008; and as deputy program manager of the Navy Tactical Networks Program Office from June 2008 to August 2010.

Captain Beel is a member of the Defense Acquisition Corps and is Level III certified in Program Management, Life Cycle Logistics and Production, and Quality and Manufacturing. He is a certified

Lean Six Sigma Black Belt. He led a continuous process improvement project that was awarded a California Council of Excellence California Team Excellence bronze award and was selected to compete for the American Society of Quality’s International Team Excellence Award at the 2011 World Conference on Quality and Improvement.

Captain Beel’s awards include the Meritorious Service Medal (three awards), Air Medal (individual award), Navy Commendation Medal (five awards), Navy Achievement Medal, and various unit, campaign, and service awards. He has also received the Sikorsky “Winged-S” Lifesaving Rescue Award.

A Framework for Reuse in the DoN

Randy Mactal—Mr. Mactal is a senior technical specialist with the Space and Naval Warfare, Systems Center, Pacific, in San Diego, CA. He has over 11 years of experience in systems engineering, net-centric and system-of-systems engineering, open architecture, service-oriented architecture, collaboration tools, and content management systems. He has successfully managed multiple projects, supervised hundreds of employees, and held multiple positions of leadership within his organization. Mr. Mactal is currently embedded within Program Executive Office, Command Control, Communications, Computers, and Intelligence (PEO C4I) where he is the project manager for the Net-centric Enterprise Solutions for Interoperability (NESI) and the C4I Domain Open Architecture Action Officer. He is a graduate of the University of Southern California and a former naval intelligence officer. [randy.mactal@navy.mil]

Lynne Spruill—Ms. Spruill has over 25 years of extensive systems engineering experience. She has been highly successful for 17 years as the proprietor of a woman-owned small business within the DoD and intelligence community. She has supported PEO C4I in San Diego for the last 10 years with expertise in systems engineering and integration, DoD acquisition and contract planning, management of data rights of intellectual property, software reuse, open architecture and net-centricity, service-oriented architecture (SOA), SOA security architectures, collaboration tools, software repositories, and the development of a SPAWAR-wide content management system. Lynne has been involved with developing and supporting the PEO C4I Reuse Framework for the past seven years. [lynne.spruill.ctr@navy.mil]

Abstract

Reuse offers the possibility of increasing engineering productivity, efficiency, and software quality while simultaneously reducing the cost of building software-intensive systems. The application of reuse has been around for many years and the DoD has made concerted efforts to implement reuse strategies since the early 90s. Although there are many excellent examples of its implementation throughout the Navy, efforts to implement software reuse strategies at an enterprise level have not matured enough to reap large-scale benefits. In the current fiscal climate of budget reductions and mandates for efficiencies, changes in acquisition, engineering, and business processes will require an enterprise reuse strategy that provides clear guidance, incentives, and compliance mechanisms. This paper discusses the current state of reuse within the DoN and proposes an implementation framework for a strategy-driven reuse approach.

Introduction

An environment of fiscal austerity driven by a decade of war and an imperative for deficit reduction is driving the DoN to implement changes and mechanisms that promote approaches that lead to better effectiveness, efficiency, and affordability in how we develop our products. In addition, changes in the technological landscape with the continued transition to service-oriented architectures (SOA), cloud computing, and increased emphasis on system-of-systems engineering is also driving the need for change. While there will be many approaches to address this fiscal crisis and technological change, this paper focuses on an approach that addresses both simultaneously, the need for a strategy-driven, enterprise approach to reuse. Reuse offers the possibility of increasing engineering productivity, efficiency, and software quality while simultaneously reducing the cost of building software-intensive systems. The application of reuse has been around for many years and the DoD has made concerted efforts to implement reuse strategies since the early 90s. Although there are many excellent examples of its implementation throughout the DoN, efforts to implement reuse strategies at an enterprise level have not matured enough to reap large-scale benefits. A strategy-driven, enterprise approach to reuse must address the change required in acquisition, engineering, and business processes to provide clear guidance, incentives, and compliance mechanisms. This paper identifies the barriers to the

success of an enterprise reuse strategy and recommends an implementation framework to address those barriers and mature our current state of reuse so large-scale benefits can be realized.

Reuse Defined

Definitions of reuse have varied throughout the years and originally focused on reuse of software code. Over time, the definition of reuse has broadened to include not only software code, but many other related artifacts and assets as shown in Table 1.

Table 1. Reuse Artifacts and Assets

Architectures	Development documents
Contracting documents	Test and evaluation plans
Contracting language	Training plans
Acquisition documents	Cost estimates
Design/development tools	Testing tools

For the purpose of this paper, reuse is defined as “the systematic use of existing assets and artifacts in the development of other software with the goal of improving productivity, efficiency, and quality to reduce costs and delivery cycle times.” In addition to this definition, it is important to consider two very important characteristics to determine an asset or artifact’s value: (1) usability, which assesses the extent to which an artifact is easy to use, regardless of its functionality; and (2) usefulness, which is the extent to which a reusable asset will often be needed, regardless of its packaging. These characteristics will be explored further in later sections of this paper.

Benefits & Challenges

Benefits

In adopting any change, each organization must analyze the benefits of that change and how that change adds value to achieving an organization’s goals. As mentioned in the Introduction, today’s environment of budget austerity will require organizations to implement changes and mechanisms that promote approaches that lead to better effectiveness, efficiency, and affordability. Implementation of a stronger reuse approach would allow the Navy to meet the requirements set forth by the budget authorities and program sponsors. Specifically, the benefits of reuse are as follows:

- Accelerated Development—Less custom development can provide a head start to development and result in speeding up delivery.
- Gains in Productivity—Associated with accelerated development, less custom development also allows resources to be used in other areas and increases productivity.

- Increased Quality/Trustworthiness of the Product—Establishing and adhering to quality standards for reusable assets, combined with prior testing, can increase confidence that the asset will perform as advertised.
- Cost Reduction/Avoidance—Less custom development can result in a direct reduction in costs. Quality reuse assets have typically already gone through testing so a reduction in testing requirements should result in cost avoidance.
- Lower Risk of Program Failure—Due to the above benefits, reuse can create less uncertainty in schedule, costs, and product quality, thus resulting in lowering the risk of program failure.

The benefits to reuse can be summarized as "*less, shorter, fewer, lower*," which translates to less custom development, shorter development cycles, fewer errors, and lower cost.

Challenges

Implementation of a stronger reuse approach does have its challenges. Identifying and understanding those challenges allows organizations to develop strategies and options for addressing those challenges. The challenges to implementation of reuse are listed in Table 2.

Table 2. Challenges to Reuse Implementation

Category	Challenge
Organizational/cultural	Organizational silos Resistance within the organization Lack of management support and focus Lack of supporting policies and standard processes Not invented here/protecting our business mentality
Economic/business	Cost of change implementation/maintenance Lack of incentives/rewards Vendor business models /vendor lock Intellectual property rights
Technical	Lack of training and technical skills Lack of standard development processes Developing quality reusable assets Incompatible software design No central repositories and discovery mechanisms

While many of the challenges listed in Table 2 may seem somewhat daunting, they are by no means insurmountable. The proposed reuse framework that will be introduced in the Framework for Reuse section of this paper provides a road map for addressing these challenges.

Reuse Levels

There are four levels of reuse that go from low maturity (ad hoc) to the highest level of maturity (strategy driven). Understanding reuse levels is imperative so that organizations can assess their level of maturity and implement the changes necessary to mature to their objective level.

Ad Hoc Reuse

Ad hoc reuse primarily occurs within project/program level efforts focused on their individual needs versus those of the organization. The project/program determines where reuse would be beneficial and implements reuse, primarily through code reuse or reuse of documentation.

Systematic Reuse

Systematic reuse occurs when a project, program, or organization includes reuse as a consideration during planning stages. In addition to consideration during planning, systematic reuse will also include dedicated processes that incorporate reuse considerations in a product's lifecycle. Reuse at this level has matured beyond software code to include reuse of other related assets and artifacts, and will usually have a supportive infrastructure.

Domain-Oriented Reuse

Domain-oriented reuse is similar to systematic, but is focused on a particular domain, such as Command, Control, Communications, Computers, and Intelligence (C4I). Additional analysis has been conducted at the higher levels to determine which reuse assets or artifacts would provide the most value to the domain, and policies or processes are in place to ensure programs in the domain are held accountable for reuse.

All three of the above levels of reuse occur throughout the Navy and there are many excellent examples of its implementation, but efforts to implement reuse strategies at an enterprise level have not matured enough to reap large-scale benefits. To reap the large-scale benefits that are possible with reuse, the Navy needs to drive the organization towards a strategy-driven reuse approach.

Strategy-Driven Reuse

Strategy-driven reuse includes characteristics of systematic and domain-oriented reuse with the addition of two important characteristics: incorporation of reuse into the organization's strategic decision-making process and structuring the organization to optimize reuse.

As with all major change efforts, a move towards a strategy-driven reuse approach should reap some short-term benefits, but to truly reap the most value over time, organizations must be diligent and patient in their implementation. Long-term success will emerge as the organization's reuse strategy matures and reuse becomes a standard practice.

Maturity of Government Reuse

Over the last decade, government reuse has steadily matured in three key areas: policy, technology, and acquisition. This shift was accelerated in the 2002–2007 timeframe with the introduction of the NII Checklist and MOSA policies. The main goals of these policies were focused on a development style that promoted openness and modularity, respectively. These policies, coupled with the emergence of collaboration tools and service-oriented development, started the shift away from building stovepipe applications to building modular services that were loosely coupled, interoperable, and reusable.

In the 2007–2009 timeframe, the acquisition area began to mature with the introduction of the *Naval Open Architecture Contract Guidebook for Program Managers* (Defense Acquisition University [DAU], 2011). The guidebook provided contracting language to help ensure software interoperability, reusability, maintainability, extensibility, and scalability. This guide also provided standard data rights labeling for government-owned

intellectual property (IP) and CDRLs for capturing the correct packaging for any lifecycle artifact. The technology change in this period was the use of a software repository to capture, discover, and share government-owned IP and the use of collaboration tools to develop software similar to the open source community.

In the 2009–2011 timeframe, the reuse paradigm again experienced gains in technology. The increased use of open source software development, services, collaboration sites, software repositories, and new discovery mechanisms has resulted in increased interaction among organizations that was not available in the past. While this technology expansion was occurring, the acquisition community continued to evolve as more policies and education emerged to address this expansion. The contracting community has also evolved as the DoD acknowledged the importance of open architecture and released a draft version of the *Open Systems Architecture (OSA) Contract Guidebook for Program Managers* (DoD, 2011), which is largely based on the Navy version but includes additional input from the other Services. Maximizing reuse is a fundamental principle of OSA. As information dominance, cloud computing, and system-of-systems engineering emerge as the driving forces for the future, reuse strategy and technology needs to evolve as well.

Reuse-Related Policies, Better Buying Power, and System-of-Systems Engineering

Reuse-Related Policies

In order to mature the current level of reuse, Navy-level policies focused on implementation of reuse must be developed. Current policies, strategies, and compliance mechanisms are woefully inadequate, and there are no Navy policies or strategies that directly focus on reuse. Typically, reuse is included within acquisition policies where implementation of Naval Open Architecture (OA) is included.

For example, SECNAV INST 5000.2E, DoN Implementation and Operation of the Defense Acquisition System and Joint Capabilities Integration and Development System states,

> Naval open architecture precepts shall be applied across the Naval Enterprise as an integrated technical and business approach and shall be used for all systems, including support systems, when developing an acquisition strategy per ASN (RD&A) memorandum, Naval Open Architecture Scope and Responsibilities, of 5 August 2004 and CNO memorandum Ser N6N7/5U916276, Requirement for Open Architecture (OA) Implementation, of 23 Dec 2005.

Further examination of the CNO memorandum mentioned in the previous quotation identifies OA principles, including the following related to reuse: “Identify or develop reusable application software selected through open competition of ‘best of breed’ candidates, reviewed by subject matter expert peers and based on data-driven analysis and experimentation to meet operational requirements.”

In another example, the draft DoD *OSA Contract Guidebook for Program Mangers* released in December 2011, contains the strongest and most robust language related to reuse as part of implementing OSA. The following are examples of that language:

> “Enterprise investment strategies, based on collaboration and trust, that maximize reuse of proven system designs and ensure we spend the least to get the best.”

> "Execution of an effective OSA strategy including strategic asset reuse must be considered from both a Pre-Award and Post-Award perspective."

In addition, Naval OA requirements were part of DoN enterprise architecture (EA) compliance measures, which were developed to support the certification processes of the various Investment Review Boards and the Defense Business Systems Management Committee. In relation to reuse, DoN EA simply asked the question of whether or not a program has established an asset reuse strategy, providing no amplifying information or guidance. Inclusion of Naval OA within DoN EA compliance measures stood as the primary mechanism for assessing OA compliance. Unfortunately, OA requirements within DoN EA have recently been suspended for the remainder of FY2012 pending further analysis.

While the references in this section provide some guidance for reuse implementation, stronger direction and compliance mechanisms related to reuse are needed in order to mature the organization to higher level so as to reap long-term benefits.

Better Buying Power

The Better Buying Power (BBP) initiatives that emerged within the last few years were created to obtain greater efficiency and productivity in defense spending. Specific actions were grouped into five major areas:

- Target Affordability and Control Cost Growth—reduce redundancy;
- Incentivize Productivity and Innovation in Industry—use fixed-price incentive firm target (FPIF) contracts for production;
- Promote Real Competition—encourage competition, reuse, and data rights;
- Improve Tradecraft in Services Acquisition—better define requirements and increase use of small business entities; and
- Reduce Non-Productive Processes and Bureaucracy—reduce redundancy in standard processes by utilizing standard system processes.

Implementation of a strategy-driven reuse approach with its associated benefits would directly support achieving greater efficiency and productivity, thus complying with the actions set forth in the BBP Memorandum (Better Buying Power Initiative, n.d.).

System-of-Systems Engineering (SoSE) and Reuse

System-of-systems (SoSE) engineering is defined in the DoD *Defense Acquisition Guidebook* (*DAG*) as, "A set or arrangement of systems that results when independent and useful systems are integrated into a larger system that delivers unique capabilities."

Traditional systems engineering seeks to optimize an individual system, while SoSE seeks to optimize a network of various interacting legacy and new systems together to satisfy multiple objectives of the program. Technical management of architectures and interfaces is crucial to effective SoSE to intrinsically design interoperability into the SoS. With interoperability as a major goal for SoSE, a more robust reuse strategy would aid the SoS manager in determining the common components of the SoS and making those components discoverable, accessible, and available to reuse.

SoSE could implement a service-oriented architecture (SOA) approach for building capability. SOA promotes loose coupling, modularity, a standards-based approach, interoperability, agile development, composeability, extensibility, scalability, and maintainability, all of which are related in some fashion or another to reuse.

Framework for Reuse

This paper proposes a need to adopt a strategy-driven reuse approach using the implementation framework defined in the following sections to address the challenges mentioned in the Challenges section of this paper. This approach will allow us to mature our current state of reuse so large-scale benefits can be realized, as depicted in Figure 1.

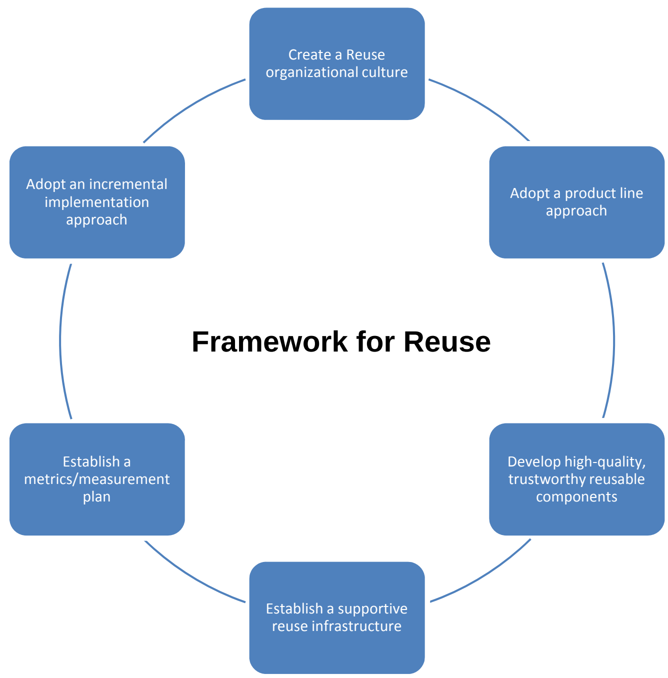


Figure 1. Framework for Reuse

Create an Organizational Culture for Reuse

Establish Stakeholder Buy-In

Before an organization decides to implement any major change, the proposed change must first demonstrate value to the organization. Stakeholders must be properly educated on the concept of reuse, its benefits and challenges, and the motivation for its implementation. Reuse champions must be designated to establish stakeholder buy-in so that at least a commitment from an organization's leadership allows further investigation. This should be followed by an examination of the feasibility of the reuse approach and the value it could bring to the organization. Presentation of the results of the feasibility analysis should result in the decision of whether or not to pursue.

Institute Supportive Policies, Processes, and Practices

While more robust naval-level policies focused on reuse would provide the impetus for achieving large-scale benefits, each organization within the DoN will still be responsible for instituting its own policies, processes, and practices to ensure a strategy-driven reuse approach is successful.

Organizations must be responsible for translating higher level policies and requirements into concrete actions. Those actions must result in making reuse processes and practices an organization's standard way of conducting business. In order to achieve strategy-driven reuse, organizations must incorporate reuse into the organization's strategic decision-making process and structure the organization to optimize reuse.

Many organizations already have policies that govern their engineering processes and practices. Within Space and Naval Warfare (SPAWAR), Systems Command, an effort to standardize system engineering processes was started over two years ago. This effort has led to the development of the *SPAWAR Systems Engineering Guide* (*SSEG*; SPAWAR, n.d.). The *SSEG* is a web-based collection of systems engineering processes and guidance that supports Team SPAWAR's mission, promotes a consistent and common view of systems engineering (SE) across Team SPAWAR, and provides an SE framework across a product's lifecycle. This is a perfect example of where reuse processes and practices should be integrated so that they become standard in the organization.

Successful implementation of reuse also goes beyond engineering. An organization's contracting processes and practices must also take reuse into consideration. Fortunately, this is one area where there is robust guidance in the form of the draft DoD *OSA Contracts Guidebook for Program Managers* (DoD, 2011). Reuse language is present throughout the guidebook, providing good direction during all phases of acquisition. While contracting guidance for reuse is readily available, it will still be up to each organization to institute the oversight necessary to ensure that this guidance is actually used.

Another important consideration related to reuse is ensuring that each organization diligently exercises its rights to the technical data and computer software procured through the development of its products. It would be easy to assume that this is standard practice, but there are many examples available that demonstrate poor execution or inattention to this consideration, resulting in increased cost to the government. By diligently exercising their data rights, organizations increase their ability to control developed assets and artifacts, which can, in turn, be reused for other development or provided to other parties for the purposes of increasing competition.

Educate & Train the Organization on Reuse

Introducing change to an organization is always a challenge and is even made more difficult when education and training are not primary considerations during the change's implementation. Proper education and training on the reuse implementation is one of the most important methods of communicating and is critical to success. Successful implementation of a strategy-driven reuse approach involves all levels of an organization, both vertically and horizontally. Decision-makers, engineers, government contractors and finance personnel, and DoN contractors all must be educated and trained on the reuse approach and the organizational objectives since they will all be involved with its implementation.

Decision-makers must be thoroughly educated and supportive of reuse so as to champion the effort and provide guidance, support, and resources, as well as institute policies and processes that aid in implementation. Engineers must be educated on how to design and develop for reuse and receive technical training on software development methods related to reuse. Government contractors and finance personnel must be educated on the higher level policies related to reuse and the use of the DoD's *OSA Contract Guidebook* (DoD, 2011) and its implications to their role in the acquisition process. DoN contractors must be educated on how reuse implementation may affect their engineering activities and business models, and must adapt accordingly. Finally, all involved in the reuse implementation must be and trained on how to utilize the supporting reuse infrastructure in the development of their products.

Education and training of the organization can be accomplished in many ways, and it will be up to each organization to determine the best approach to accomplish this.

Create Incentives and Rewards That Encourage Reuse

While education and training may be used to encourage change, additional motivation via incentives and rewards may be needed as well. There are a range of incentives and rewards available that can be used to encourage both individuals and teams, and these should not be limited to just positive reinforcement.

From a government perspective, implementation of reuse could be incentivized through the use of individual or team recognition within the organization. Most organizations already have such programs in place and should easily be able to add recognition for reuse as a category for consideration. Recognition can be made through the numerous communication methods an organization has at its disposal. Cash or on-the-spot awards are other examples of this type of incentive. Incentives could also come through organizational recognition of the importance of reuse through developing specialized training and through emphasis on on-the-job enrichment through professional development.

It is also important to recognize that incentives do not always have to be positive, since you can tie accountability to the individual's performance objectives. Program/project managers are responsible for overseeing successful product implementations in terms of cost and schedule and, therefore, should be held accountable for implementation of reuse.

From a contractor perspective, incentives are designed to motivate contractor performance that might not otherwise be emphasized. Incentives for reuse are built around cost, schedule, management, data rights, and technical merits. Each incentive can apply a different emphasis on using percentages to evaluate performance. Incentivizing technical excellence in the program is an important aspect of the program acquisition strategy and is usually applied with award fees or award terms. Incentives can be structured around the following principles:

- linking award fees to acquisition outcome,
- linking award term contacting extensions,
- limiting the opportunities for earning unearned fees in subsequent periods (fee rollover),
- designing evaluation criteria to motivate excellent performance, and
- not paying for unsatisfactory performance.

Instead of rewarding the contractor with additional fees for exceptional performance, reward the contractor by extending the contract period of performance in the form of additional term periods added on to the basic contract. Under an award term incentive, the government monitors and evaluates the contractor’s performance, and if it is decided that the contractor’s performance was excellent, then the contractor earns an extension. During subsequent evaluations, if the contractor maintains excellent performance, additional terms are awarded. Contract extensions can last as long as 10 years.

In addition, contractors’ business models must evolve and adapt to the changes occurring in the acquisition landscape. Better Buying Power initiatives are beginning to take hold and contractors must recognize that to be competitive, they must also incentivize their own organizations to evolve and adapt.

Adopt a Product-Line Approach

The Software Engineering Institute defines a product line as, “A set of software-intensive systems that share a common, managed set of features satisfying the specific needs of a particular market segment or mission and that are developed from a common set of core assets in a prescribed way.”

Product-line managers have typically conducted upfront analysis to determine commonality/variation points. A result of that analysis is a defined set of core reusable assets that can be used for the development of other assets or new products. Common architectures, software, documentation, and interfaces are the main examples of assets that can be reused across the product line. Effective implementation of a product-line approach results in very similar benefits to those achieved by implementing reuse: improved productivity, increased quality, decreased time to market, and decreased cost. An effective product-line approach takes full advantage of reuse and, conversely, an effective reuse approach should take advantage of a product-line approach.

Develop High-Quality, Trustworthy Reusable Components

Component-based engineering is a style of software engineering focused on designing and composing new capabilities from reusable assets. The term *component* is defined as reusable assets that are self-contained and independent, require little customization (plug-n-play), and provide well-defined services to the applications in which they integrate.

Component-based software engineering (CBSE; also known as component-based development [CBD]) is a branch of software engineering that emphasizes the separation of concerns in respect to the wide-ranging functionality available throughout a given software system. It is a reuse-based approach to defining, implementing, and composing loosely coupled independent components into systems. The idea of loose coupling relates to how tightly the behavior of one component is bound to the implementation details of other components.

Software engineers regard components as part of the starting platform for service orientation. An individual software component is a software package, a web service, or a module that encapsulates a set of related functions (or data). All system processes are placed into separate components so that all of the data and functions inside each component are semantically related. Because of this principle, it is often said that components are modular and cohesive. The idea of high cohesion relates to the behavior of a single component. A highly cohesive component encapsulates one concept—it does one thing well. Highly cohesive modules are easier to understand, easier to maintain, and easier to reuse. On the contrary, low-cohesive components try to do too much. They try to encapsulate too many concepts. They tend to grow exponentially and, over time, become more complex and more difficult to maintain.

With regard to system-wide co-ordination, components communicate with each other via interfaces. When a component offers services to the rest of the system, it accesses a standards-based interface that specifies the services that other components can utilize, and how they can do so.

When developing good reusable components, these additional requirements are used in conjunction with the design characteristics described previously:

- *Well documented*. It should be documented for reuse including any integration documentation.
- *Useful.* The component should demonstrate its value and usage.
- *Certified and Secure*. All components should be verified to network access.
- *Supports government ownership/data labeling.* All government reusable assets need proper labeling with handling instructions.

- *Discoverable.* Metadata about each component that is a service should be made available for discovery purposes.
- *Deposited in software repository.* All government-owned assets should be electronically scanned, then deposited into the software repository.

Establish a Supportive Reuse Infrastructure

Organizations must establish a supportive reuse infrastructure that provides the governance, processes, and tools necessary to support reuse throughout a product's lifecycle. The four main functions of a reuse infrastructure are depicted in Figure 2.

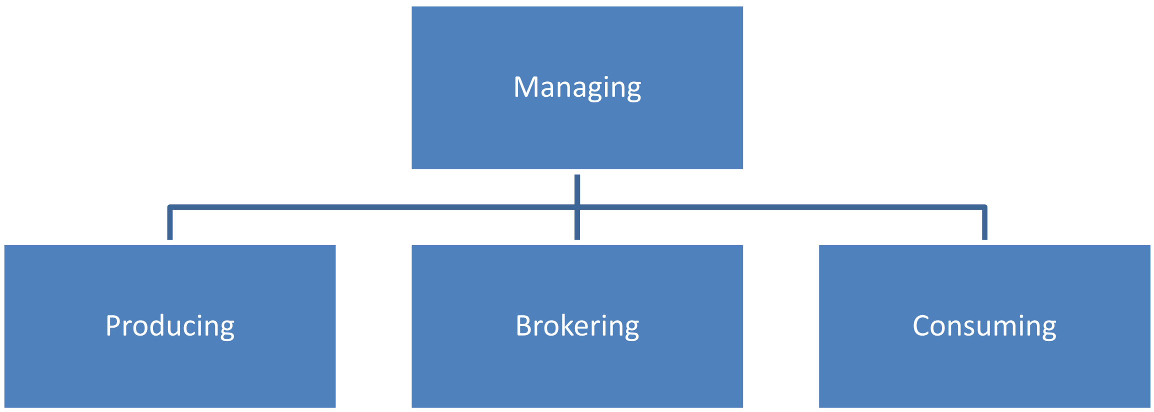


Figure 2. Functions of a Reuse Intrastructure

Managing the Reuse Infrastructure

This function focuses on providing the governance and planning necessary to properly direct and administer the reuse infrastructure. Coordinating, resourcing, and assessing the performance of the reuse infrastructure are primary responsibilities in this function. In addition, this function will be responsible for determining what reuse assets and artifacts are of value to the organization, and for determining and applying strict quality criteria for asset or artifact acceptance into the infrastructure.

Producing Reusable Assets

This function focuses on providing the processes and tools necessary to produce and maintain reusable assets and artifacts. The use of a collaboration site for development is critical to the success of a reuse infrastructure since it provides a central development hub and, ideally, provides basic functionality, such as document management, requirements and configuration management, tracker mechanisms, and chat or forums for collaboration and discussion.

In addition to the basic functionality, additional tools beneficial to the reuse process should be incorporated into a tool suite integrated with the collaboration site. Recommended tools include the following:

- software code quality analysis,
- intellectual property rights markings scan,
- verification and validation, and
- contractual documentation scan.

A library/repository is also required for storing and maintaining important assets and artifacts, such as software, documentation, specifications, business processes, etc. An ideal library/repository would have the capability to provide tiered information display, metrics collection, and discovery mechanisms.

Brokering Reusable Assets

This function focuses on how assets are procured, certified, added, or removed from the library/repository. It serves as the main interface with both external and internal users and will involve the implementation of a discovery/registry mechanism—used to catalog and search for metadata about each component in the repository. This discovery mechanism will interface with the repository so searches for reusable assets can be done easily.

It should be recognized that efforts to create an enterprise-wide repository have met with limited success in the Navy, but this should not deter organizations from implementing their own repositories, as long as attention is paid to other existing repositories and a federated discovery capability is built into them.

Consuming Reusable Assets

This function focuses on how available assets are used to develop other assets or improve existing systems. Consumers use the discovery mechanism to search the repository for available assets, complete the necessary procedures for retrieving those assets, and then use those assets for their specific needs. Another important role that consumers play in this function is providing feedback to the infrastructure manager on the usability and usefulness of the infrastructure and the assets they retrieved. This feedback loop assists in continuous improvement to the infrastructure and enhances future use.

Reuse Lifecycle Process

The reuse lifecycle is made up of a combination of business processes that include acquisition, contracting, and technical activities. The lifecycle for a reusable asset spans the need to add language in the solicitation phase, perform an assessment in the development phase, and capture deliverables in the engineering phase. See Table 3 for details.

Table 3. Process Lifecycle of a Reusable Asset

Lifecycle Event	Purpose	Activities
Solicitation phase	To get the bidder community to consider reuse of existing government IP	Add reuse language to sections of solicitation
		Attach CDRLs to contract
		Provide e-commerce instructions to bidder community
		Check for existing artifact in repository to assess reusability
		Bidder response provided whether reuse will be successful
Technology development phase	To assess the artifact is properly package for reuse and to deliver any government IP to the repository	Run technical assessment to evaluate openness and integrate component
		Deposit deliverable and packaging to repository for acceptance
Engineering and production phase	To apply data rights labeling to newly composed or developed capability, and deposit in repository	Apply data labels to software header file and documentation
		Scan artifact, product report for analysis
		Make available for discovery

Reuse Metrics

Reuse metrics and models are used to improve productivity and quality that aligns to the Dr. Carter Better Buying Power memo (Better Buying Power Initiative, n.d.). Reuse metrics help define the measure(s) for monitoring and controlling quality goals for processes and products. Reuse can apply to any lifecycle product, but is mostly used with software engineering. This section introduces five metrics, shown in Table 4, that could be used as measures for successful reuse. While the first three metrics show an increase in adoption of reuse, the last two provide cost savings by reducing redundancy.

Table 4. Metrics to Measure Successful Reuse

Metrics	Description	Tool
OA Reusability Assessment Metrics	Planning for reuse early in program lifecycle	DITPR/DoN database OA Compliance Rule Set
Maturity Metrics	What artifacts are mature enough to reuse	Technical assessment and scan tool (data labels)
Repository Usage Metrics	Number of programs using software repository in solicitations and sharing software and documentation	NESI Collaboration Site SHARE
Software Metrics	Software Lines of Code (SLOC) count	Standard SLOC formula
Service Metrics	Number of programs reusing services	Cost-avoidance formula
Software Quality	Helps with STRs and CASREPS	Scan tools and software inspection

OA Reusability Assessment Metrics

This metric is defined by programs using reuse in acquisition strategies (AS). The number is represented as a percentage and is calculated using information from the DITPR DoN database and OA compliance assessments. The percentage is calculated based on the number of programs that have responded, divided by the total number of programs. This is a compliance percentage and indicates how well many programs are using reuse as a part of their strategic direction.

Maturity Metrics

This metric is used to describe the OA tenets for modularity, interoperability, maintainability, extensibility, composeability, and reuse. Software components and documentation are technically assessed to evaluate openness, and scan tools are used to ensure proper data rights markings are applied correctly to reusable artifacts. Both are used for artifacts that get deposited in the software repository.

Repository Usage

This metric is defined by the number of programs using the site to expose government-owned IP to the community as a part of their solicitation process. Additional metrics, such as the number of downloads and the assets that are most often downloaded, should also be considered.

Software (SLOC) Metrics

Source a line of code (SLOC) is a software metric used to measure the size of a software program by counting the number of lines in the text of the program's source code. SLOC is typically used to predict the amount of effort that will be required to develop a program, as well as to estimate programming productivity or maintainability once the software is produced (Wikipedia).

Service Metrics

Service metrics are measured by the cost avoidance associated with a reduction in duplicate services, and in consolidation and streamlined processes. Cost avoidance includes those costs that would have been incurred if reuse were not implemented. Avoided costs come threefold: first, deduction in product lifecycle costs and manpower. The second avoided cost is shortened development schedules for services providers integrating into existing platforms. For service reuse to work, a common framework needs to be used. For component-based frameworks, this platform is service-oriented architecture (SOA). The third avoided cost is reduction in the time-to-market and manpower by reusing paperwork required for certificate and accreditation (C&A) to securely certify components in the network.

Software Quality

Software quality measurement is about identification of discrete critical programming errors using a combination of scan tool and inspections. These vulnerabilities are the result of bad practices that under specific circumstances can lead to catastrophic outages, performance degradations, security breaches, corrupted data, and myriad other problems that make a given system unsuitable for use, regardless of its rating based on aggregated measurements. The measurement of critical application characteristics involves measuring structural attributes of the application's architecture, coding, in-line documentation, and proper data rights labeling.

Adopt an Incremental Approach

Implementing change is often a challenge to most organizations and evidence of successful change management shows that adopting an incremental approach for any major organizational change increases the chances for success. It is recommended that the implementation of the reuse framework described in the previous sections should also adopt an incremental approach, as described in Figure 3.

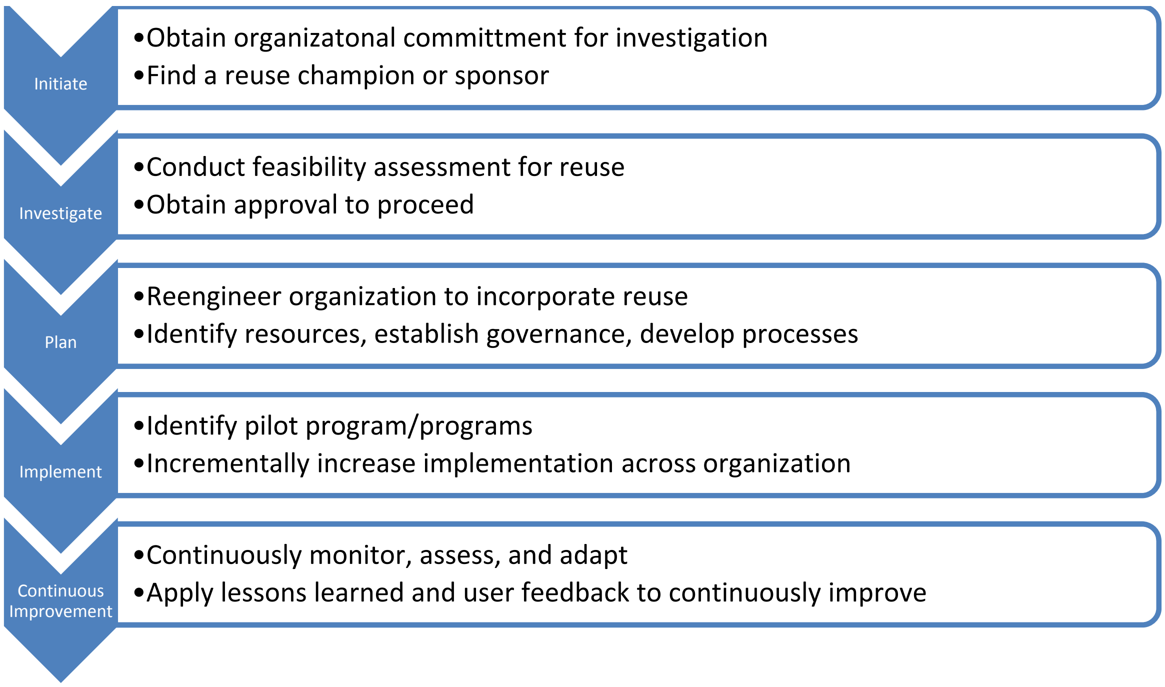


Figure 3. Incremental Approach to Reuse Implementation

The depiction of the phases in Figure 3 serves as an overview for the recommended incremental approach for reuse implementation. Each phase will have a basic structure that identifies phase goal, personnel involved, specific activities for each phase, and performance measures. Adopting an incremental implementation will allow the reuse strategy to mature and continuously improve over time.

Conclusion

The Navy must implement approaches that lead to better effectiveness, efficiency, and affordability in how we acquire and develop our products. While there are policies and guidance that direct the implementation of open architecture, there is a need for a more robust focus on the area of reuse. Reuse offers the possibility of increasing engineering productivity, efficiency, and software quality, while simultaneously reducing the cost of building software-intensive systems. But in order to reap large-scale benefits of reuse, a strategy-driven reuse approach must be implemented.

References

Better buying power initiative. (n.d.). Retrieved from https://acc.dau.mil/bbp

Defense Acquisition University (DAU). (n.d.). Software reuse [Coursework, CLE 041].

Defense Acquisition University (DAU). (2011, December). *Naval open architecture contracting guidebook for program managers* (Version 0.1). Retrieved from https://acc.dau.mil/CommunityBrowser.aspx?id=22108

DoD. (2011, December). *Open systems architecture contract guidebook for program managers*. Washington, DC: Author.

Jacobson, I., Griss, M., & Jonsson, P. (2007). *Software reuse: Architecture, process and organization for business success*.

Lim, W. C., (1998). *Managing software reuse*.

Mili, H., Mili, A., Youcob, S., & Addy, E. (2011). *Reuse-based software engineering: Techniques, organization, and controls*.

SSC-PAC. (2011, March). *Government reuse presentation to PEO C4I acquisition community forum (ACF)* [Technical briefing].

SSC-PAC Engineering. (2012, March). *The rapid integration and test environment approach to modernization of software acquisition, source code management, and agile software lifecycle management* [Technical paper].

Space and Naval Warfare (SPAWAR). (n.d.). Technical guidance and checklist(s; Version 3.3). Retrieved from Net-Centric Enterprise Solutions for Interoperability (NESI) website: http://nesipublic.spawar.navy.mil/

ACQUISITION RESEARCH PROGRAM
GRADUATE SCHOOL OF BUSINESS & PUBLIC POLICY
NAVAL POSTGRADUATE SCHOOL
555 DYER ROAD, INGERSOLL HALL
MONTEREY, CA 93943

www.acquisitionresearch.net

Program Executive Office
Command, Control, Communications,
Computers and Intelligence (PEO C4I)

A Framework for Reuse in the DON

May 16, 2012
Randy Mactal
PEO C4I
randy.mactal@navy.mil

Statement D: Distribution authorized to Department of Defense (DOD) and U. S. DOD contractors only. Other requests must be referred to PEO C4I or the SPAWAR Office of Congressional and Public Affairs (SPAWAR 00P).

Abstract

- Reuse offers the possibility of increasing engineering productivity, efficiency, and software quality while simultaneously reducing the cost of building software-intensive systems.
- Efforts to implement software reuse strategies at an enterprise level have not matured enough to reap large scale benefits.
- In the current fiscal climate of budget reductions and mandates for efficiencies, changes in acquisition, engineering and business processes will require an enterprise reuse strategy.

Brief will propose an implementation framework for a Strategy-Driven Reuse Approach

Reuse Defined

- The systematic use of existing artifacts and assets in the development of software with the goals of:
  - Improving productivity, efficiency, and quality
  - Reducing costs and delivery cycle times
- Reuse could be applied to virtually *any* aspect of acquisition and engineering:
  - Architectures
  - Contracting documents
  - Contracting language
  - Acquisition documents
  - Design/development tools
  - Development documents
  - Test and evaluation plans
  - Training plans
  - Cost estimates
  - Testing tools

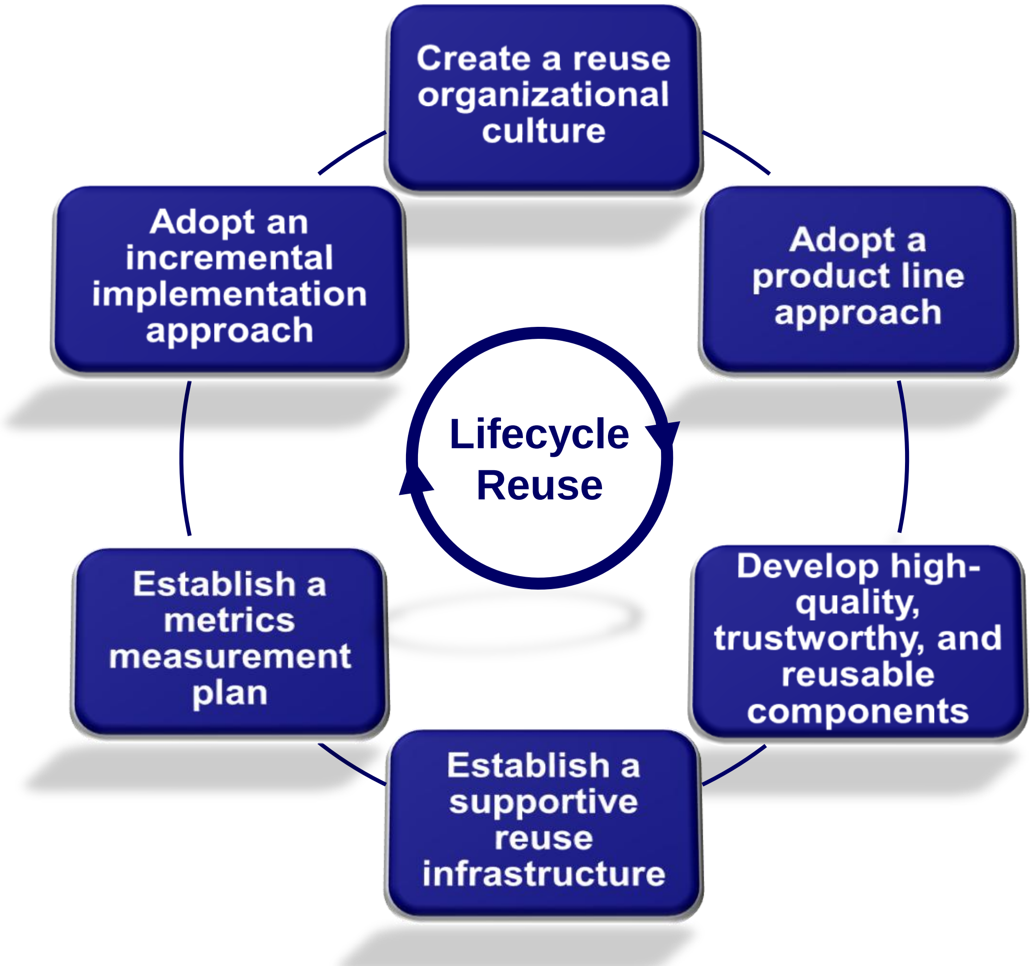
Reuse extends beyond software code

Levels of Reuse

- Ad Hoc
  - Primarily code reuse
  - Project/individual effort
- Systematic
  - Planned reuse with processes
  - Reuse of other assets beyond code
  - Typically have supportive infrastructure
- Domain-Oriented
  - Analysis to determine domain focused reuse assets
- Strategy-Driven
  - Organization structured to support and optimize reuse
  - Incorporation of reuse into strategic decision making process

Strategy-driven, enterprise approach is needed

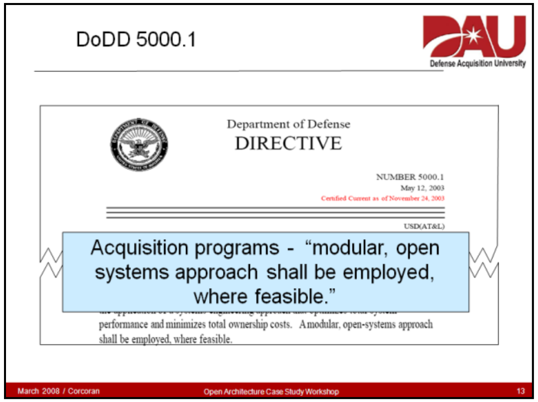
Framework for Reuse

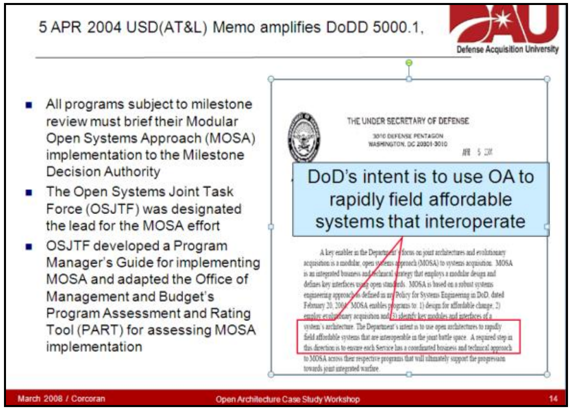


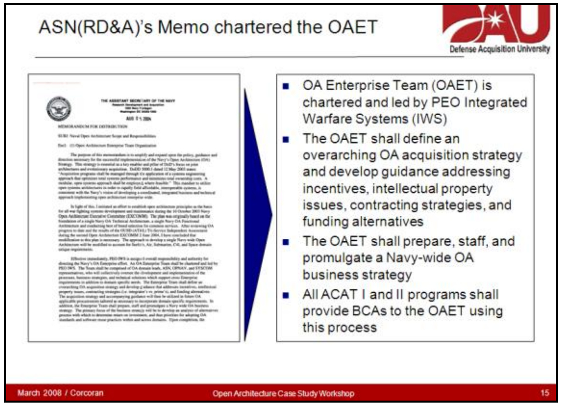
This Framework proposes the need to adopt a Strategy-Driven Reuse approach, so large scale reuse benefits can be realized in the enterprise.

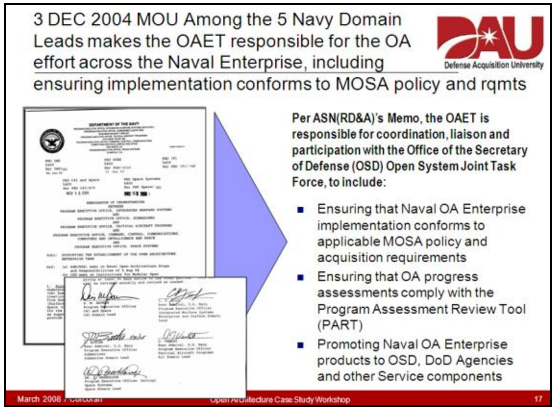
Reuse Policy

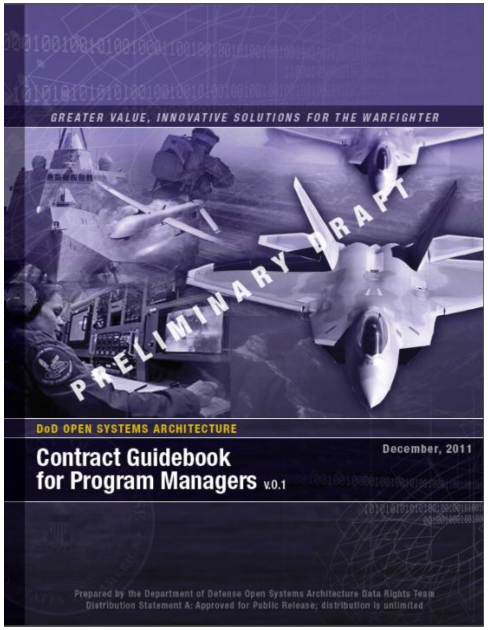
Create a reuse organizational culture







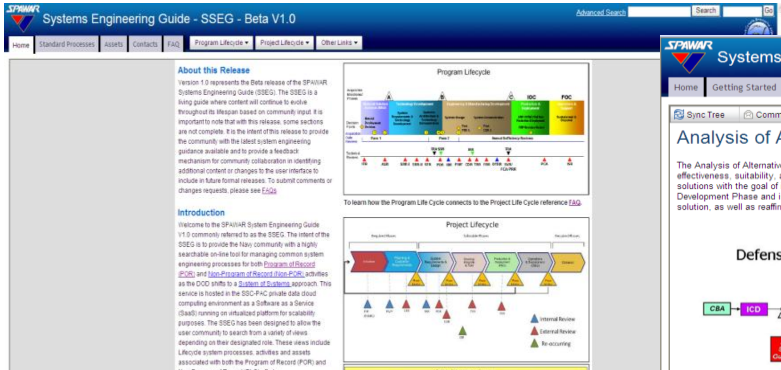


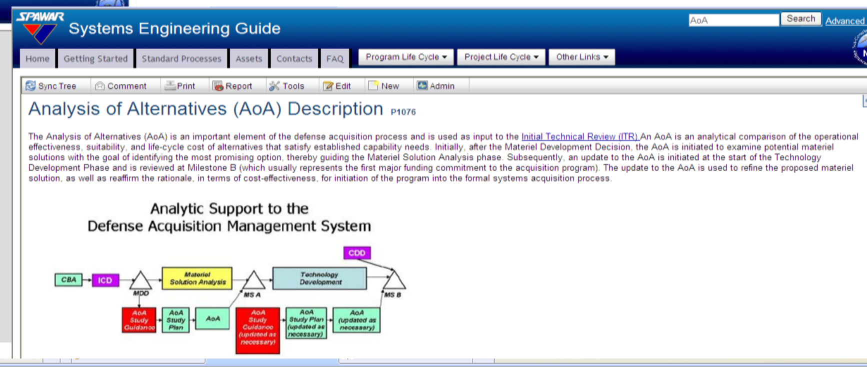


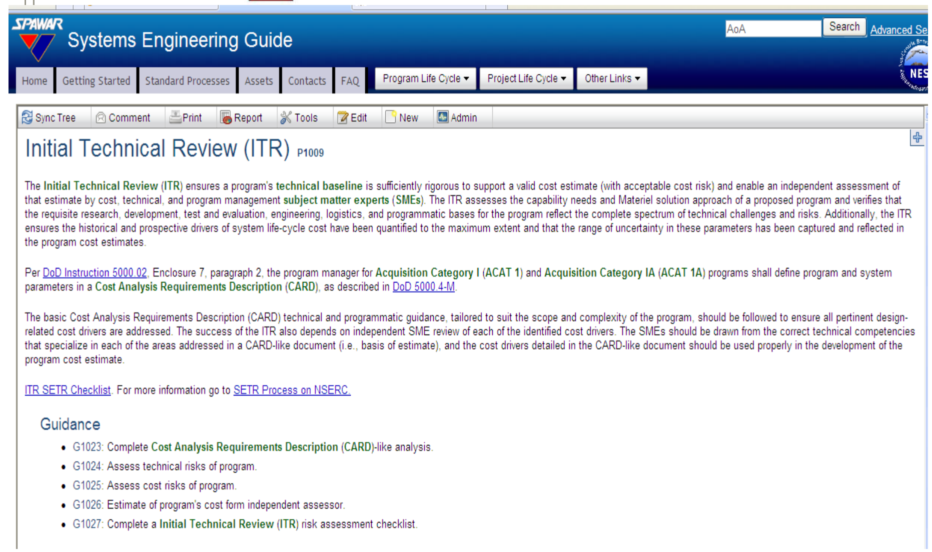
Robust Navy policy on Reuse must extend beyond OA policy

Policies, Processes, and Practices

Create a reuse organizational culture







SPAWAR 5.0 Process Asset Library (PAL)

- 5.0 HQ Standard Operating Procedures (SOP)
- 5.0 Concept of Operations (CONOPS)
- SPAWAR System Engineering Guide (SSEG)
- System Center PALs
  - SSC Atlantic PAL
  - SSC Pacific PAL

Dominance The Navy's Information Systems Command SPAWAR

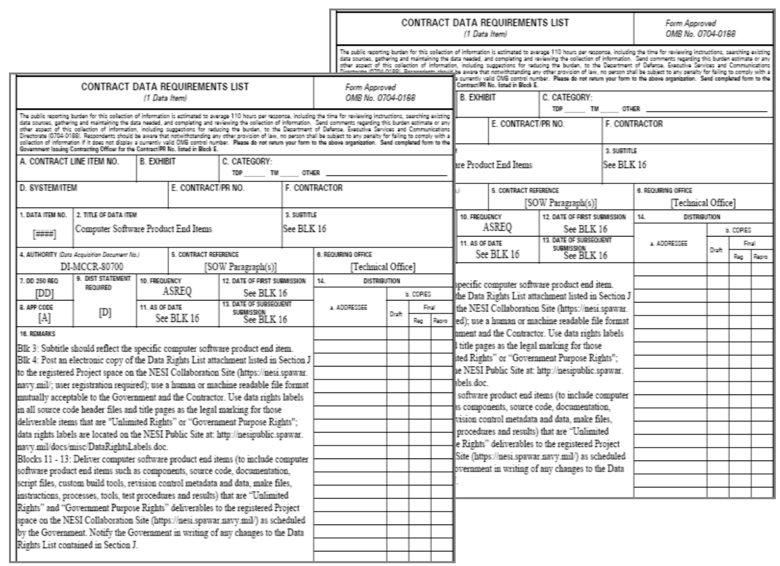
Reuse must become part of the organizational culture and established as standard processes and practices

Improved Contracting Practices

Create a reuse organizational culture

- Solicitation Language
  - **Section B** – for the Technical Data, Computer Software, Computer Software Documentation and data rights. The section provides a list of the deliverables and associated data rights in a table.
  - **Section H** – for identification of rights and restrictions. This section describes the DFARS definitions of unlimited and Government Purpose Rights (GPR).
  - **Section L** – for proposal instructions on how reuse will be measure and evaluated. The section describes the value of reuse, but doesn't mandate reuse.
  - **Section M** – for evaluation criteria & applying factors (weights). GFI response is evaluated and scored.

- Standard CDRLs
  - Insert data rights labels
  - Post deliverables to the Collaboration Site
  - Submit SLOC information associated with software reuse as a deliverable



CONTRACT DATA REQUIREMENTS LIST
(1 Data Item)
Form Approved
OMB No. 0704-0188

The public reporting burden for this collection of information is estimated to average 110 hours per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing the burden, to the Department of Defense, Executive Services and Communications Directorate (0704-0188). Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number. **Please do not return your form to the above organization. Send completed form to the Government Issuing Contracting Officer for the Contract/PR No. listed in Block E.**

A. CONTRACT LINE ITEM NO. | B. EXHIBIT | C. CATEGORY: TDP ____ TM ____ OTHER ____

D. SYSTEM/ITEM | E. CONTRACT/PR NO. | F. CONTRACTOR

1. DATA ITEM NO.: [####] | 2. TITLE OF DATA ITEM: Computer Software Product End Items | 3. SUBTITLE: See BLK 16

4. AUTHORITY (Data Acquisition Document No.): DI-MCCR-80700 | 5. CONTRACT REFERENCE: [SOW Paragraph(s)] | 6. REQUIRING OFFICE: [Technical Office]

7. DD 250 REQ: [DD] | 9. DIST STATEMENT REQUIRED: [D] | 10. FREQUENCY: ASREQ | 12. DATE OF FIRST SUBMISSION: See BLK 16 | 14. DISTRIBUTION

8. APP CODE: [A] | 11. AS OF DATE: See BLK 16 | 13. DATE OF SUBSEQUENT SUBMISSION: See BLK 16 | a. ADDRESSEE | b. COPIES: Draft, Final (Reg, Repro)

16. REMARKS

Blk 3: Subtitle should reflect the specific computer software product end item.
Blk 4: Post an electronic copy of the Data Rights List attachment listed in Section J to the registered Project space on the NESI Collaboration Site (https://nesi.spawar.navy.mil/; user registration required); use a human or machine readable file format mutually acceptable to the Government and the Contractor. Use data rights labels in all source code header files and title pages as the legal marking for those deliverable items that are "Unlimited Rights" or "Government Purpose Rights"; data rights labels are located on the NESI Public Site at: http://nesipublic.spawar.navy.mil/docs/misc/DataRightsLabels.doc.
Blocks 11 - 13: Deliver computer software product end items (to include computer software product end items such as components, source code, documentation, script files, custom build tools, revision control metadata and data, make files, instructions, processes, tools, test procedures and results) that are "Unlimited Rights" and "Government Purpose Rights" deliverables to the registered Project space on the NESI Collaboration Site (https://nesi.spawar.navy.mil/) as scheduled by the Government. Notify the Government in writing of any changes to the Data Rights List contained in Section J.

Increase competition, reduce cost, and increase buying power

Legal Labeling

Data Rights Labels, Warranty Notices, Disclaimers

Create a reuse organizational culture

Unlimited

//

/// **SECURITY CLASSIFICATION: UNCLASSIFIED**

//

Copyright (C) *<Date>*, *<company name>*

Notwithstanding any copyright notice, U.S. Government rights in this work are defined by DFARS 252.227-7013 or DFARS 252.227-7014 as detailed below. Use of this work other than as specifically authorized by the U.S. Government may violate any copyrights that exist in this work.

/// UNLIMITED RIGHTS

/// DFARS Clause references: 252.227-7013 (a)(15) and 252.227-7014 (a)(15)
/// Unlimited Rights. The Government has the right to use, modify, reproduce, perform,
/// display, release or disclose this (technical data or computer software) in whole or in part, in
/// any manner, and for any purpose whatsoever, and to have or authorize others to do so.
///
/// **Distribution Statement D.** Distribution authorized to the Department of Defense and
/// U.S. DoD Contractors only in support of US DoD efforts. Other requests shall be
/// referred to SPAWAR PMW xxx.
///
/// **Warning:** - This document contains data whose export is restricted by the Arms Export
/// Control Act (Title 22, U.S.C., Sec 2751, et seq.) as amended, or the Export Administration
/// Act (Title 50, U.S.C., App 2401 et seq.) as amended. Violations of these export laws
/// are subject to severe criminal and civil penalties. Disseminate in accordance with
/// provisions of DoD Directive 5230.25.

THE GOVERNMENT DOES NOT WARRANT, INCLUDING THE WARRANTIES OF MERCHANTABILITY AND OF FITNESS FOR A PARTICULAR PURPOSE,THE QUALITY OR COMPLETENESS OF THE SOFTWARE LISTED IN THE .PDF DOCUMENT ENTITLED [**insert program name**] MASTER GFI LIST"POSTED ON THE [**insert program name**] VENDOR INFORMATION NESI WEBSITE. THIS SOFTWARE AND SUPPORTING DATA ARE BEING PROVIDED FOR INFORMATION ONLY, TO BE USED FOR THE LIMITED PURPOSE OF THE [**insert program name** SOLICITATION ONLY (N00000-00-X-0000). ACCESS TO THIS DATA FOR PURPOSES OF THIS SOLICITATION DOES NOT AUTHORIZE THE OFFEROR TO USE, REPRODUCE, OR SHARE THIS DATA FOR ANY OTHER PURPOSES. THE OFFEROR SHALL NOTIFY AND GAIN CONSENT FROM THE [**insert program name**] PCO PRIOR TO ANY OTHER PROPOSED USE, EITHER GOVERNMENT OR OTHERWISE.

Labeling Used In IP

The [program name] materials in the data packages located on this site for the unit level system are Government licensed Intellectual Property (IP) received with SBIR Data Rights. The Government has rights under any copyright subsisting in this data. Upon expiration, SBIR Data Rights become unlimited rights wherein the Government has the right to use, modify, reproduce, perform, display, release or disclose this (technical data or computer software) in whole or in part, in any manner, and for any purpose whatsoever, and to have or authorize others to do so.

Distribution Statement B. Distribution authorized to U.S. Government agencies only. Other requests for distribution shall be authorized by SPAWAR SSC PAC/ SPAWAR HQ PCO [PMW TPOC notified].

Warning: - This document contains data whose export is restricted by the Arms Export Control Act (Title 22, U.S.C., Sec 2751, et seq.) as amended, or the Export Administration Act (Title 50, U.S.C., App 2401 et seq.) as amended. Violations of these export laws are subject to severe criminal and civil penalties. Disseminate in accordance with provisions of DoD Directive 5230.25.

Access to this information constitutes understanding and acceptance of this disclaimer.

Labeling used on Site

Government must diligently exercise their intellectual property rights

Industry Incentivization

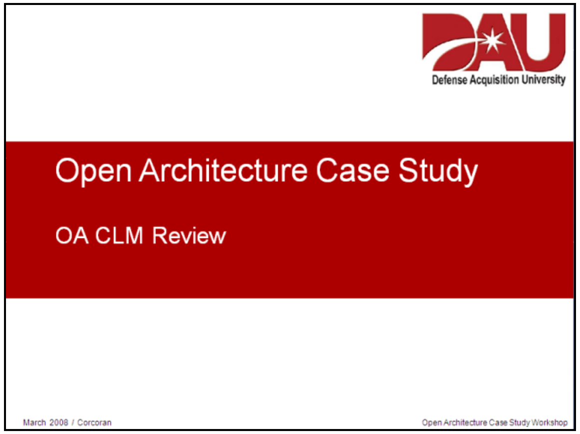
Create a reuse organizational culture

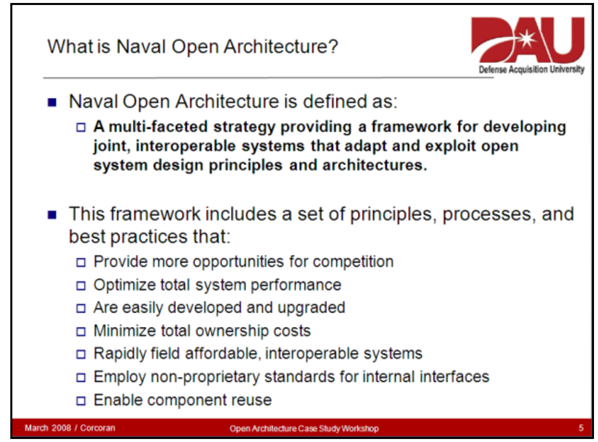
- Adopting reuse requires changes to business models
- Incentives are built around Cost, Schedule, Management, Data Rights and Technical merits
  - Each can support a different emphasis using percentages
- Incentives include:
  - Linking award fees to acquisition outcome
    - Contract extensions (10 years max timeline, is possible)
  - Limiting the opportunities for earning unearned fees in subsequent periods (fee rollover)
  - Designing evaluation criteria to motivate excellent performance
  - Not paying for unsatisfactory performance

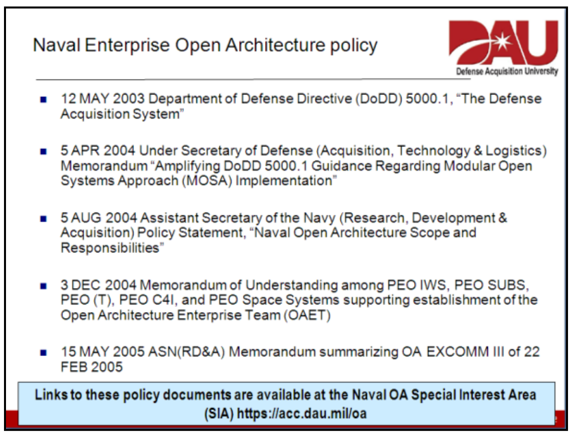
Incentives must be incorporated in order to affect change in business models

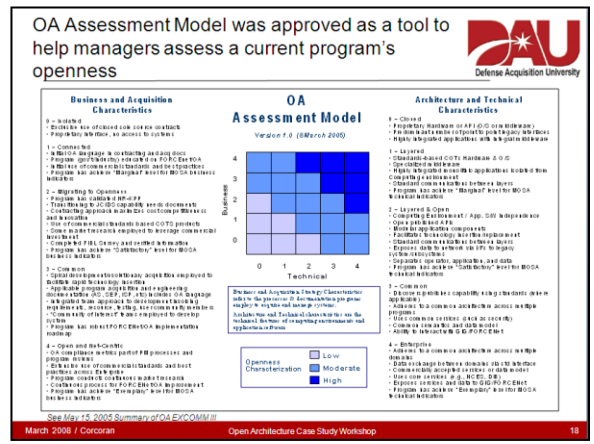
Educate the Organization

Create a reuse organizational culture







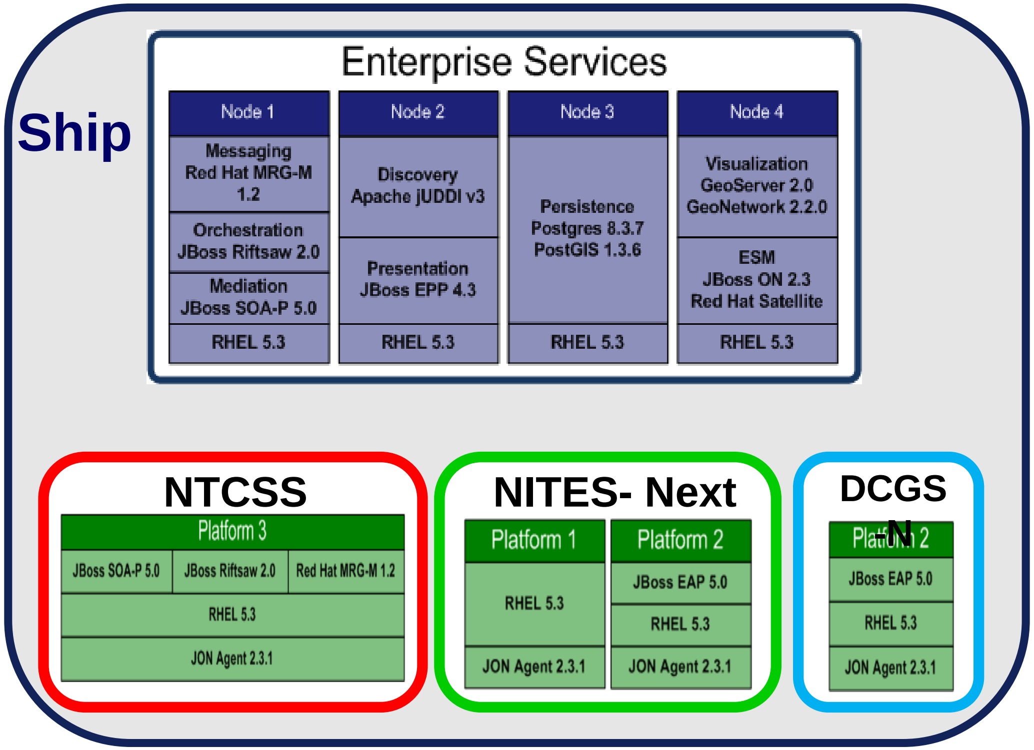


Organizations must educate PMs, engineers, contractors, and others on Reuse

Product Line Development

Adopt a product line approach

- Common, managed set of features that are developed from a common set of core assets.
- Improvements in time to market, cost, productivity, and quality.



Reuse and product line development are tightly coupled

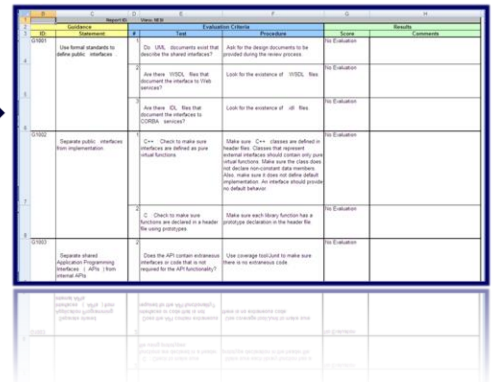
Building Quality, Reusable Components

Develop high-quality, trustworthy, and reusable components

1. Add Reuse Contracting language to Solicitation

Solicitation Language

- **Section B –** for the Technical Data, Computer Software, Computer Software Documentation and data rights. The section provides a list of the deliverables and associated data rights in a table.
- **Section H –** for identification of rights and restrictions. This section describes the DFARS definitions of unlimited and Government Purpose Rights (GPR).
- **Section L –** for proposal instructions on how reuse will be measure and evaluated. The section describes the value of reuse, but doesn't mandate reuse.
- **Section M-** for evaluation criteria & applying factors (weights). GFI response is evaluated and scored.



3. Perform incremental technical assessments
4. Well documented
5. Modular & Cohesive
6. Loosely coupled
7. Defined interfaces
8. Composable

Documentation

Software

9. Apply Data Labeling
10. Perform Quality and Data Rights Scan

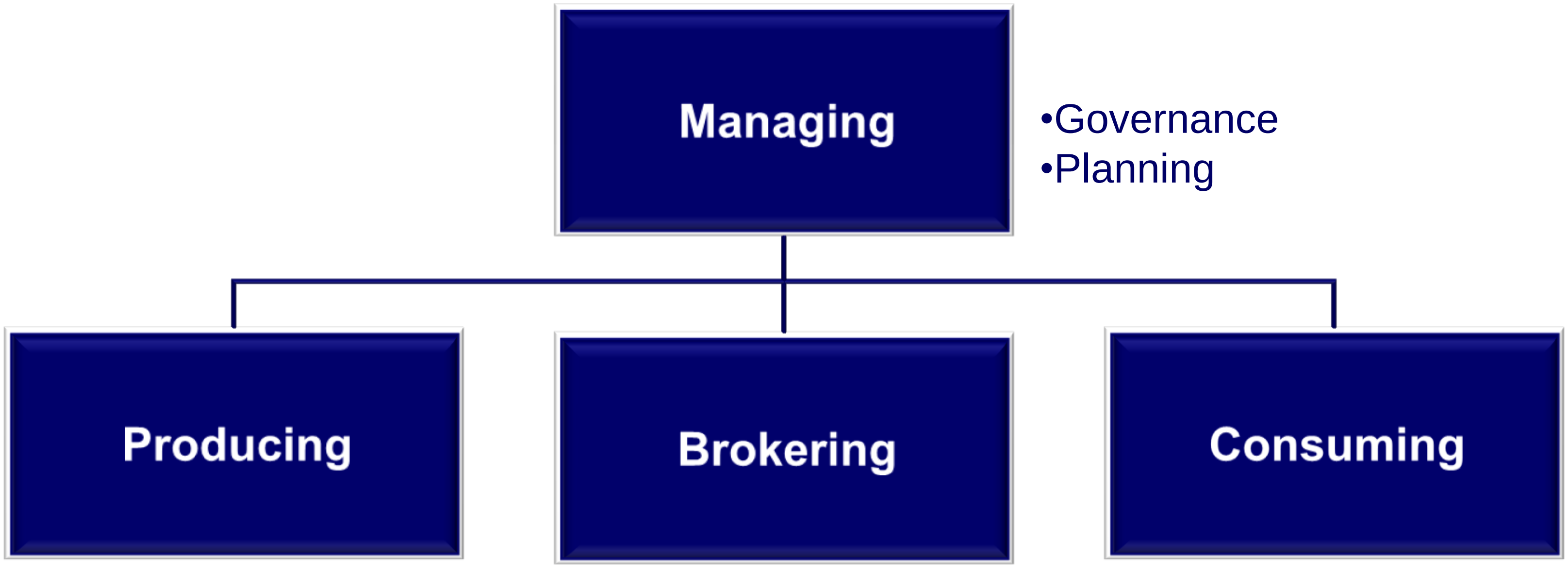
Metadata Service Registry ↔ Software Repository

11. Deposited in software repository
12. Make discoverable - Registered in service registry

Organizations must "design and build for reuse"

Supportive Reuse Infrastructure

Establish a supportive Reuse infrastructure



- Governance
- Planning

- Processes
- Standards
- Tools

- Procurement
- Certification
- Added, or removed from the library/repository

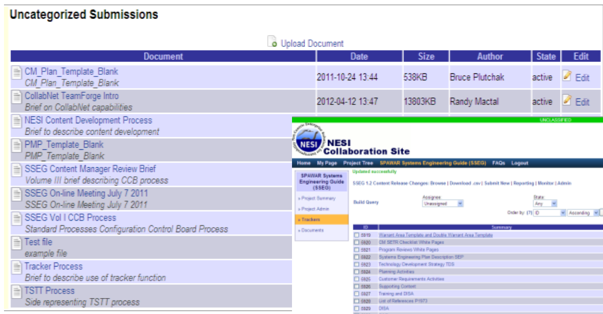
- Asset availability
- Asset Composeability

Provides governance, processes, and tools to support reuse throughout a product's lifecycle

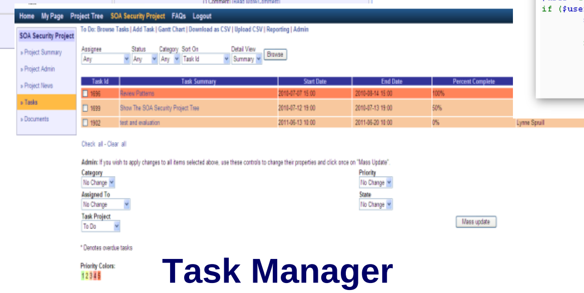
Current infrastructure capabilities

Establish a supportive Reuse infrastructure

Document Manager



File Release System

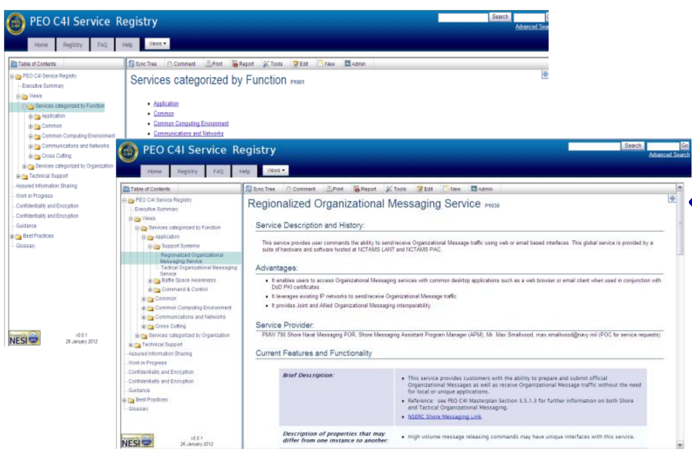


Source Code Manager

Task Manager

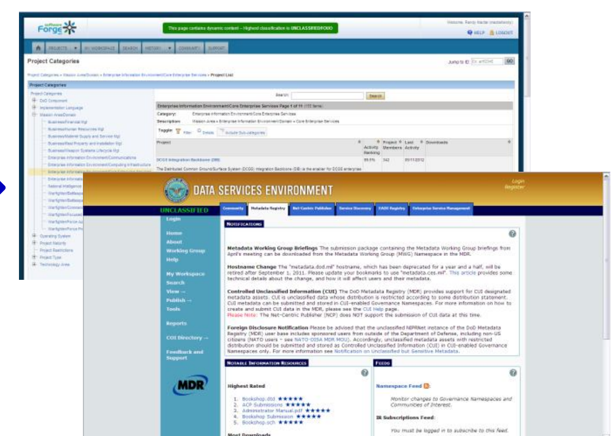
Trackers

Internal Discovery



Federated

External Discovery



Technical infrastructure requires extensive integration

Ideal Technical Reuse Infrastructure Environment

Establish a supportive Reuse infrastructure

Central Administration

- Discover
- Federate
- Replicate

COLLABNET Subversion Edge

Central Repository

Lifecycle Traceability

Automation for End-to-End Processes

Plan	Code	Track	Build and Test	Release
User/Project Admin Task Hierarchy	Browse, Tracker SCM Integration	Bugs, Artifacts Requirements	Integrated Hudson	Collect, Archive, Release

- Build
- Test
- Run

Lab Management

Team Collaboration

Real-time Reports & Status

Discussion Forums

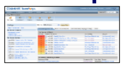
Project-based Wiki

Indexed Objects 300+ File Types

Social Architecture

Community & Projects

My Workspace



Tasks Tracking

Project List

Monitor

Settings

User Tool

Business Users
CollabNet Desktop, Microsoft Windows, Project

Developers
CollabNet Desktop, Eclipse Edition, Visual Studio Edition

User Point Tools
Lifecycle Tools, e.g., Code Analysis, Unit Test, Load Test, Report Management, Planning

ScrumWorks Pro

Infrastructure must be agile, composeable, and scalable

Reuse Metrics

Establish a metrics measurement plan

METRICS	Description	Tool
Reusability Assessment	Planning for reuse early in program lifecycle	OAAT, SETR, Gate Reviews
Maturity	What artifacts are mature enough to reuse	Technical assessment & scan tool (data labels)
Repository usage	Number of programs using software repository in solicitations & sharing software and documentation	NESI Collaboration Site, Discovery tools
Software	Software Lines of Code (SLOC) count	Standard SLOC formula
Service	Number of programs reusing services	Cost avoidance formula
Quality	Helps with STRs & CASREPS	Scan tools and software inspection

Drive improvement, direct focus, improve decisions

Adopt an Incremental Approach

Adopt an incremental implementation approach

Initiate
- Obtain organizational commitment for investigation
- Find a reuse champion or sponsor

Investigate
- Conduct feasibility assessment for reuse
- Obtain approval to proceed

Plan
- Reengineer organization to incorporate reuse
- Identify resources, establish governance, develop processes

Implement
- Identify pilot program/programs
- Incrementally increase implementation across organization

Continuous Improvement
- Continuously monitor, assess, and adapt
- Apply lessons learned and user feedback to continuously improve

Change can be disruptive, smooth the transition

Conclusion

- The Navy must implement approaches that lead to better effectiveness, efficiency, and affordability in how we acquire and develop our products.
- Reuse offers an approach to address those needs.
- A more robust focus on the area of reuse that extends beyond OA is needed.
- In order to reap large scale benefits of reuse, a Strategy-Driven Reuse approach must be developed.
- Organizations must develop a cohesive implementation framework to be successful.

"If you do what you've always done, you'll get what you've always gotten." ***Tony Robbins***

Contact Info

Randy Mactal
Net-Centric Engineering & Integration
SPAWAR Systems Center, Pacific
PEO C4I
53560 Hull St.
San Diego, CA 92152
e-mail: randy.mactal@navy.mil
Work: 858-537-8944

Lynne Spruill
APEO Engineering Support
PEO C4I
4301 Pacific Highway
San Diego, CA 92110
e-mail: lynne.spruill.ctr@navy.mil
Cell: 619-985-6266

We get IT.

We also integrate it, install it and support it. For today and tomorrow.

Visit us at www.peoc4i.navy.mil

Examples of Ad-Hoc Reuse

Program	Service Reuse
DCGS-N Incr II	Reusing Afloat Core Services (ACS), Ozone Widget Framework (OWF), & GCCS-M Components
NITES-Next	Reusing ACS & DCGS-N services, GCCS-M 4.1 OGC services, the OWF widgets being created by FNMOC METOC C2RPC efforts
NTCSS	Reusing ACS, developing reusable business processes
G-TSCMIS	Reusing I-TSCMIS software & DECC shared services
CANES - ACS	Building reusable core services to include: Messaging, Discovery, OWF, BPEL, Application Server
ADNS Incr III	Reused software to create services Incr III capabilities
GPNTS	ACS, refactoring NAVSSI software into services

- Ad-hoc reuse involves:
  - ➢ refactoring existing software into services
  - ➢ use of CANES SOA foundation (ACS)

Strategy-driven reuse requires a Framework

Acronyms

ACS: Afloat Core Services

ADNS: Automated Digital Network System

BPEL: Business Process Execution Language

C2RPC: Command & Control Rapid Prototype Capability

CANES: Consolidated Afloat Network Enterprise Services

CASREP: Casualty Report

CCE: Common Computing Environment

CDRL: Contract Data Requirements List

DCGS-N: Distributed Common Ground System – Navy

DECC: Defense Enterprise Computing Centers

DITPR: DoD IT Portfolio Repository

FNMOC: Fleet Numerical METOC Center

GCCS-M: Global Command and Control System – Navy

GFI: Government Furnished Information

GPNTS: GPS-based, Positioning, Navigation, and Timing Service

GPR: Government Purpose Rights

GPS: Global Positioning System

IP: Internet Protocol

METOC: Meteorological and Oceanographic

NAVSSI: Navigation Sensor System Interface

NESI: Net Centric Enterprise Solutions for Interoperability

NITES: Naval Integrated Tactical Environmental System

NTCSS: Naval Tactical Command Support System

OA: Open Architecture

OGC: Open Geospatial Consortium

OWF: Ozone Widget Framework

PAL: Process Asset Library (SPAWAR)

PEO: Program Executive Office

POR: Program of Record

RHEL: Red Hat Enterprise Linux

SCM: Source Code Manager

SLOC: Software Lines of Code

SOA: Service-Oriented Architecture

SPAWAR: Space & Naval Warfare Systems Command

STR: Software Trouble Report

TSCMIS: Theater Security Cooperation Management Information System (I – Interim, G – Global)